nature methods

BRIEF COMMUNICATION

https://doi.org/10.1038/s41592-018-0069-0

Quanti.us: a tool for rapid, flexible, crowd-based annotation of images

Alex J. Hughes[1,2,8,9], Joseph D. Mornin[3,9], Sujoy K. Biswas[2,4], Lauren E. Beck[5], David P. Bauer[2,6], Arjun Raj[5], Simone Bianco[2,4] and Zev J. Gartner[1,2,7]*

We describe Quanti.us, a crowd-based image-annotation platform that provides an accurate alternative to computational algorithms for difficult image-analysis problems. We used Quanti.us for a variety of medium-throughput image-analysis tasks and achieved 10–50× savings in analysis time compared with that required for the same task by a single expert annotator. We show equivalent deep learning performance for Quanti.us-derived and expert-derived annotations, which should allow scalable integration with tailored machine learning algorithms.

Image analysis is increasingly crucial in quantitative biology and medicine[1,2]. Features in images can be accurately annotated by humans, but this approach becomes impractical when one is working with hundreds to thousands of images. Therefore, researchers use custom-written scripts to analyze images via methods such as contrast-based segmentation, edge detection, and tracking[2–4]. These approaches can give excellent performance under specific experimental conditions, but they can respond unpredictably to slight variations in experimental setup. Issues related to image volume and diversity have motivated the development of machine learning algorithms including convolutional neural networks that are trained on extensive sets of human-annotated images[5]. Synthetic datasets offer researchers some ability to circumvent high annotation burdens, but they often do not capture the range of real-world phenotypes that algorithms must discriminate between[6,7].

Crowdsourcing offers an attractive alternative. Indeed, the scientific community has begun to build annotation pipelines that leverage large groups of human annotators working in parallel. Specific large-scale image-annotation projects have been custom-built in platforms such as EyeWire[1] and Project Discovery[8]. Zooniverse[9] aims to make crowd annotation accessible to scientists across disciplines by allowing researchers to select from a palette of image-annotation tools, and offers a modular interface for use by volunteer annotators. However, all of these platforms rely on a volunteer labor pool, which requires continuous marketing or 'gamification' to draw attention to individual projects, and to make up for inconsistent motivation among human annotators and temporal volatility in volunteer numbers[10,11]. Practically speaking, this means that jobs can suffer from lower annotation collection rates and quality[12] (Supplementary Note 1). Other crowdsourcing approaches, such as Amazon's "Mechanical Turk," enable operators to circumvent these problems through the use of micropayments. However, services like Mechanical Turk are not yet used extensively by the life sciences community, perhaps because they lack interfaces for generic image annotation and have not been quantitatively validated.

We therefore developed Quanti.us, a flexible portal that helps scientists recruit groups of untrained Mechanical Turk workers ('Turkers') to annotate images using a set of interaction tools that can be applied individually to many types of jobs. Annotations can be collected and used in series to refine further rounds of annotation, like pre-segmented input to conventional algorithms, or used as training data for machine learning algorithms (Fig. 1a).

The Quanti.us website allows researchers to upload image sets, choose an analysis tool, and provide simple sets of instructions. Turkers are presented with individual images or sets of sequential images as stacks via a 'slider' interface. Each image or stack is referred to as a 'task' within a larger 'job'. The website automatically interfaces with Mechanical Turk to set up tasks and return raw data to the researcher. These data include click location, Turker identification numbers, and time stamps. Quanti.us can also provide a link to users for access to a free 'test mode' that allows them to bypass Turkers and recruit annotators from other communities such as classrooms, the general public, or research groups (Methods).

We first evaluated Quanti.us for a particle-discrimination task involving images of fluorescent cells migrating through a porous Transwell membrane (Fig. 1b). Counting such cells on the basis of contrast-defined segmentation is difficult because the autofluorescent pores of the membrane are hard to distinguish from cells. We tasked Turkers with clicking on cells with a crosshair tool and evaluated their performance relative to a 'ground truth' expert dataset. We found that 59% of Turkers who completed at least one image performed better in terms of both precision (the ability to exclude false positives) and recall (the ability to exclude false negatives) than a semi-automated FIJI pipeline consisting of brightness threshold, watershed, and particle-size threshold steps (Fig. 1c).

We asked multiple Turkers to analyze each image and then leveraged the 'wisdom of crowds' to improve the overall performance by means of two strategies[13]. First, subtractive spatial clustering of the annotations from ten replicate Turkers produced precision and recall metrics of 0.99 and 0.81—0.015 and 0.18 higher, respectively, than the values obtained from application of the performance envelope followed by the FIJI algorithm for different particle-size thresholds. The ability of crowds to mitigate the effects of rare poorly performing workers was accrued for as few as three replicates per image

[1]Department of Pharmaceutical Chemistry, University of California, San Francisco, San Francisco, CA, USA. [2]NSF Center for Cellular Construction, University of California, San Francisco, San Francisco, CA, USA. [3]Independent Researcher, Berkeley, CA, USA. [4]Department of Industrial and Applied Genomics, IBM Accelerated Discovery Laboratory, IBM Almaden Research Center, San Jose, CA, USA. [5]Department of Bioengineering, University of Pennsylvania, Philadelphia, PA, USA. [6]Department of Cellular and Molecular Pharmacology, University of California, San Francisco, San Francisco, CA, USA. [7]Chan Zuckerberg Biohub, San Francisco, CA, USA. [8]Present address: Department of Bioengineering, University of Pennsylvania, Philadelphia, PA, USA. [9]These authors contributed equally: Alex J. Hughes and Joseph D. Mornin. *e-mail: zev.gartner@ucsf.edu

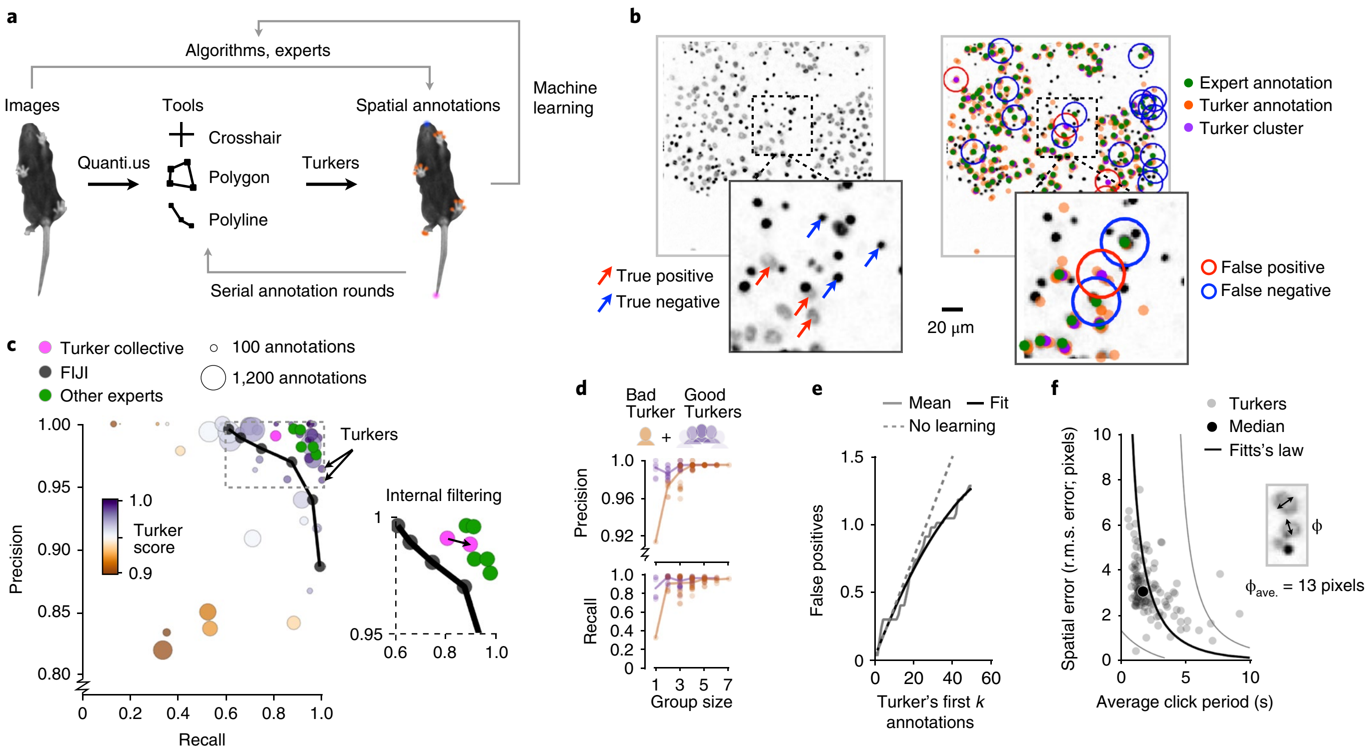

Fig. 1 | Leveraging the wisdom of crowds for scientific image analysis with Quanti.us. a, Scientists designate a tool that human Turkers then use to annotate uploaded images according to a set of brief instructions. The resulting annotations can be interpreted in raw form and used as input to conventional algorithms, or be used as training data for machine learning algorithms. **b**, Left, raw example image of cell nuclei (true positives) and autofluorescent pores (true negatives). Right, corresponding overlay of expert, Turker, and clustered Turker crosshair annotations. False positive and false negative annotations were scored against those provided by a trained expert for individual Turkers, or for spatially clustered annotations from all Turkers (Methods). Each of 300 images was annotated by ten Turkers (a subset of 20 images was used to determine Turker performance). The scale bar applies to the higher-magnification (bottom) images, which represent the regions outlined by dashed squares in the corresponding images above; high magnification is 3× that in the lower-magnification image. **c**, Precision and recall metrics for individual Turkers ($n=46$), for the clustered annotations from ten Turkers completing each image ("Turker collective"), for other experts not involved in ground truth annotation, and for a conventional FIJI object-detection pipeline over a range of particle-size thresholds. An inherent Turker quality score is shown. The gray dashed box indicates the portion of the graph highlighted in the inset to the right. Inset: the arrow indicates the effect of filtering out the bottom one-third of workers, assessed in terms of their performance, on the basis of this score. **d**, Annotations from every combination of a representative set of one to six 'good' Turkers and one 'bad' Turker who completed the same five image tasks were clustered and used to determine the indicated performance metrics. **e**, False positive errors contributed over the first *k* annotations submitted by a Turker (in chronological order), fit by a quadratic function ($n=29$ Turkers). **f**, Spatial error of annotations versus the time between annotations, with Fitts's law tradeoff ($n=129$ Turkers). Fit envelopes are 95% confidence intervals. Data are representative of two experimental replicates.

(Fig. 1d). Second, we generated an inherent Turker quality score by comparing annotations from each Turker with clusters generated from annotations made by their peers, which could allow more poorly performing Turkers to be automatically screened out, even without an expert dataset for comparison[14]. When we filtered out the contributions of the bottom third of Turkers, the performance of Quanti.us increased to within the range of the performance of five other experts who were not involved in generation of the ground truth dataset (Fig. 1c, inset). Turkers also improved in performance at cell/pore discrimination over their first 50 annotations, with average false positive rates decreasing from 3.7% to 1.3% per click (Fig. 1e). These data suggest opportunities to further improve Turker performance by, for example, providing an initial training image set.

In our evaluation of pointing accuracy, we saw that average Turker clicking periods correlated with root-mean-square (r.m.s.) errors, according to Fitts's law of speed–accuracy tradeoffs in human pointing tasks[15] (Fig. 1f). Median Turker click times were ~2 s for r.m.s. errors of ~3 pixels. Overall, the 129 Turkers who made at least one annotation had r.m.s. errors of less than 13 pixels (the average diameter of the cells they annotated). In agreement with other studies of worker contributions in crowds, the number of images attempted by each Turker followed the Pareto '80/20' principle: ~20% of Turkers accounted for ~80% of images completed[10] (Supplementary Fig. 1).

Because Quanti.us pays individuals to annotate images through Mechanical Turk, Turkers change their performance and job choice on the basis of the economic tradeoffs inherent in completing a task accurately and quickly[16]. We measured the relationships among Turker performance, task complexity, overall task completion rate, and the amount paid per task. We ran calibration experiments on synthetic ground truth images containing variable numbers of spatially distributed particles. We found that Turkers tolerated around 60 annotations per image task at a pay rate of $0.02 per image (Supplementary Fig. 2a–d). Above this image-complexity threshold, the average number of images attempted by Turkers dropped from 15 to 8. At complexity levels above 110 particles per image, recall dipped from >0.8 to ~0.6, and the rate of collected annotations dropped from >25,000 h^{-1} to ~3,000 h^{-1}. However, an increase in the amount paid per image broadly reversed these complexity-associated losses in recall and overall annotation rate (Supplementary Fig. 2e–g). We observed an annotation collection rate of ~10^5 h^{-1} for images requiring 110 annotations each at $0.06 per image, reflecting a time savings of 10–50× compared with that required for a single

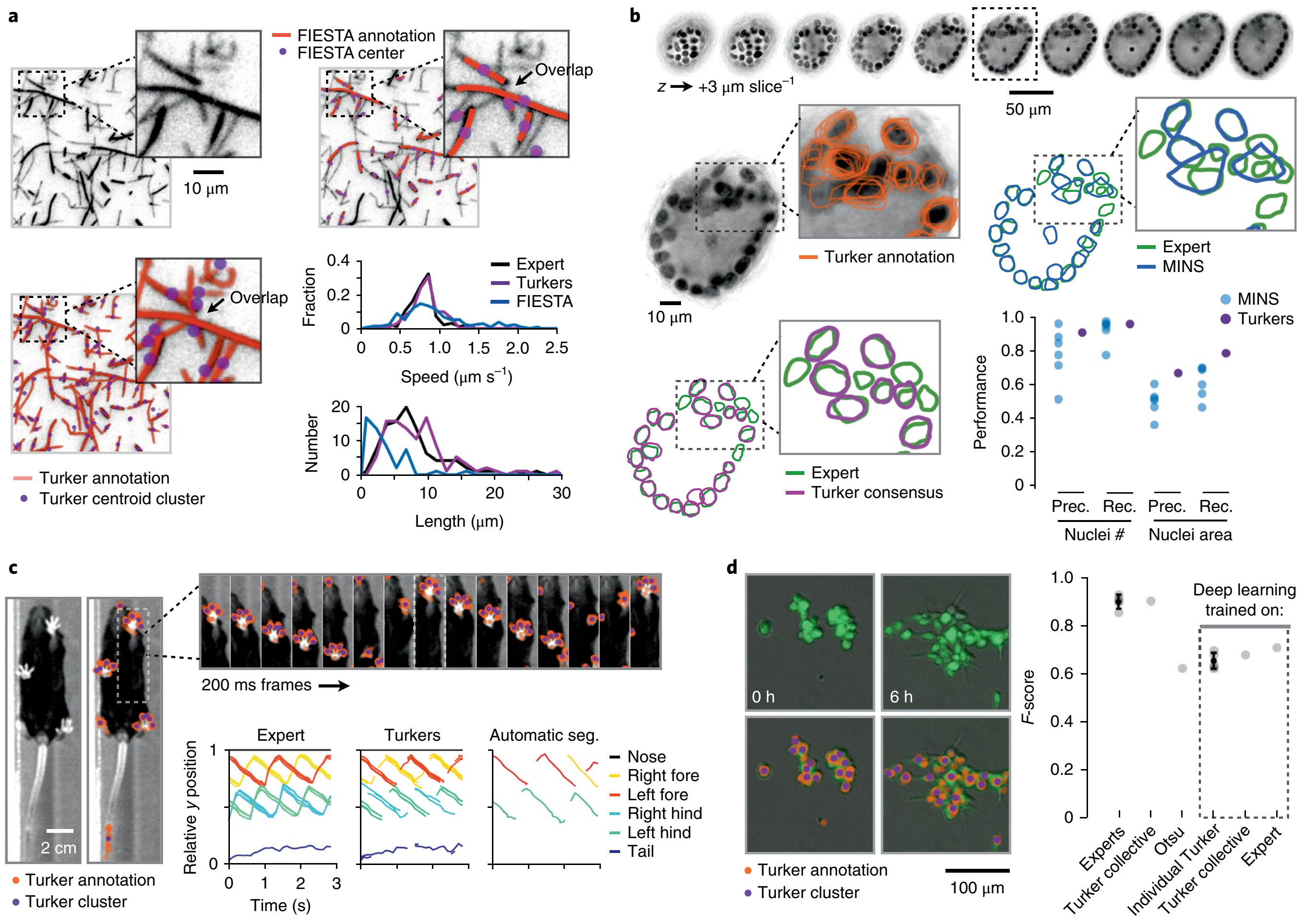

Fig. 2 | Case studies and machine learning integration of Quanti.us. a, Left, raw example image (top) and corresponding overlay of Turker annotations and clustered annotations (bottom) of fluorescent microtubules in a gliding assay, annotated with a polyline tool. Each of 50 images was annotated by ten Turkers. Right, FIESTA output (top). Plots (bottom) show microtubule speed and length distributions. The scale bar applies to the higher-magnification images, which represent the regions outlined by dashed squares in the corresponding images; high magnification is 2.75× that in the larger, lower-magnification images. **b**, Top, raw image frames of a 3D *z*-stack spanning an organoid. Middle, raw Turker outlines of nuclei, Turker consensus outlines, expert outlines, and MINS algorithm outlines associated with one frame of the stack (outlined by a dashed rectangle). Ten Turkers annotated 30 frames. The plot in the lower right shows performance metrics (prec., precision; rec., recall) for MINS for 18 runs spanning a range of parameter settings (Methods), and for the Turker collective, relative to results from an expert. **c**, Left and top, raw example images and corresponding overlays of Turker annotations and clustered annotations of the nose, digits, and tail of a walking mouse (images adapted with permission from ref. 21, Springer Nature). Each of 29 images was annotated by 20 Turkers. We input expert or spatially clustered Turker annotations into FIJI's TrackMate to construct gait plots (bottom) and also compared them to results of a conventional segmentation (seg.) pipeline in FIJI. "Hind" and "fore" refer to limbs. **d**, Left, raw example images (top) and corresponding overlays (bottom) of Turker annotations and clustered annotations for 2 of 48 frames from a movie of mammary epithelial cell spreading (ten Turkers per frame). Right, *F*-score plotted for five experts; the Turker collective; automated Otsu segmentation; and convolutional neural networks trained on annotations from five randomly chosen Turkers, clustered Turker annotations, or expert annotations. Data are shown as mean ± s.d. and are representative of at least two experimental replicates.

expert annotator to achieve similar accuracy (Supplementary Fig. 2e and Supplementary Note 2).

We next assessed the performance of this approach on more complex annotation tasks. First, we asked Turkers to draw polylines (piecewise linear curves) over microtubules recorded in a 'gliding' motility assay[17] (Fig. 2a). Such images are challenging to segment automatically because microtubules often overlap. We spatially clustered polyline annotations from ten Turkers per image, and used these cluster centers as input to FIJI's TrackMate plugin. We then compared these data with the output of a semi-automated gliding assay analysis package, FIESTA[4,17]. Although both Quanti.us and FIESTA velocity distributions approximately matched that recovered from manual microtubule tracking by an expert, the Quanti.us microtubule-length distribution matched the expert distribution more closely than FIESTA's. This seemed to be because Turkers were better at ignoring overlap junctions between microtubules, whereas FIESTA tended to break microtubule annotations into smaller segments bordered by junctions (Fig. 2a).

Pushing Quanti.us toward 3D image analysis, we asked Turkers to draw closed polygons over cell nuclei in frames from a stack of fluorescence microscopy images of epithelial cysts (Fig. 2b). We clustered outlines from individual Turkers by thresholding their degree of overlap, and generated consensus outlines suitable for comparison with outlines from a conventional 3D nuclear segmentation algorithm (MINS)[3] or from a trained expert. The consensus Turker outlines and expert outlines gave similar estimates of the number of nuclei in the stack, resulting in precision and recall metrics greater than 0.9 for the Turker collective compared with the expert's values.

Certain parameter sets used during MINS analyses gave similar performance, although these parameters required optimization to suit a particular frame in each stack. We used a pixel-wise scheme to analyze precision and recall in order to compare estimates of nuclear area[18], and observed moderate performance of the Turker collective compared with that of the expert. However, in this analysis the Turker collective performed better than MINS across a wide range of parameters. We also saw similar outlining performance for Turker collectives and experts in an epithelial organoid annotation task that required segmentation of bright-field microscopy images against a dynamic background of migrating single cells, which negatively affected the performance of an automated segmentation algorithm (Supplementary Fig. 3).

We tasked Turkers with making multiple crosshair annotations to track the nose, digits, and tail of a freely moving mouse in a movie showing the mouse's ventral aspect (Fig. 2c). Clustered Turker annotations were analyzed by FIJI's TrackMate, and successfully captured the dynamics recovered through manual gait analysis by a trained expert. These dynamics were missed by conventional contrast-based segmentation consisting of brightness thresholding, particle analysis, and TrackMate because of difficulty in distinguishing the mouse from the background. In the more difficult case of tracking ants in low-contrast images acquired near terrestrial nests (Supplementary Note 3 and Supplementary Fig. 4), we found that an initial deficiency in Turker performance compared with that of an expert for single images could be overcome if Turkers were presented with multiple images via the slider interface, which took advantage of the ants' movement to make them more easily detectable. This shows that although the Turkers lacked the prior knowledge and experience of experts, this deficit could be compensated for when the task was presented in a more tailored context.

Finally, we tested the use of Quanti.us-derived annotations as training data for machine learning. As proof of principle, we studied a movie of fluorescently labeled mammary epithelial cell clusters spreading over an in vitro culture surface, a particularly challenging problem because of frequent cell overlaps and heterogeneous cell morphologies that change over time (Fig. 2d). We trained a deep convolutional regression network[19,20] on Turker annotations to determine whether it could achieve performance similar to that of a network trained on expert annotations (Supplementary Note 4). After designing a two-stage training procedure, we produced an algorithm trained on the annotations of ten Turkers; this yielded an *F*-score (the harmonic mean of precision and recall) similar to that of an algorithm trained on the annotations of an expert (0.68 and 0.71, respectively). Both algorithms showed better performance than traditional Bayes-optimal Otsu segmentation (*F*-score of 0.62). Further, the performance of the algorithm trained on the Turker collective was better than the mean performance of algorithms trained on annotations from individual Turkers, reflecting a 'wisdom of the crowd' benefit to the training process (Supplementary Fig. 5).

Quantitation of biomedical imaging data remains a major bottleneck. The Quanti.us approach addresses this bottleneck by making crowd analysis of scientific images fast and applicable to many annotation problems. We show here that Quanti.us can enable researchers to gather hand annotations quickly, at significant scale and with high quality, by marshaling paid Turkers to annotate a range of image types. Annotations of difficult segmentation tasks may be used both for rapid pilot-scale analyses and to train convolutional neural networks. Quanti.us is also designed to allow nimble image annotation that better suits the iterative cycles of imaging, analysis, and hypothesis reformulation that characterize life science research. Pools of even higher-quality Turkers could be curated through dynamic performance tracking. Further, Quanti.us tasks could be integrated with machine learning to produce multi-stage annotation pipelines. These efforts would simplify quantitative biology analyses for fundamental, health-related, and diagnostic ends.

Methods

Methods, including statements of data availability and any associated accession codes and references, are available at https://doi.org/10.1038/s41592-018-0069-0.

Received: 25 September 2017; Accepted: 22 May 2018;
Published online: 31 July 2018

References

1. Kim, J. S. et al. *Nature* **509**, 331–336 (2014).
2. Chen, F. et al. *Nat. Methods* **13**, 679–684 (2016).
3. Lou, X., Kang, M., Xenopoulos, P., Muñoz-Descalzo, S. & Hadjantonakis, A.-K. *Stem Cell Rep.* **2**, 382–397 (2014).
4. Ruhnow, F., Zwicker, D. & Diez, S. *Biophys. J.* **100**, 2820–2828 (2011).
5. Esteva, A. et al. *Nature* **542**, 115–118 (2017).
6. Goodfellow, I. J. et al. in *Proc. Advances in Neural Information Processing Systems 27* (eds. Ghahramani, Z. et al.) 2672–2680 (NIPS, La Jolla, CA, 2014).
7. Salimans, T. et al. *arXiv* Preprint at https://arxiv.org/abs/1606.03498 (2016).
8. Thul, P. J. et al. *Science* **356**, eaal3321 (2017).
9. Simpson, R., Page, K. R. & De Roure, D. in *Proc. 23rd International Conference on World Wide Web* 1049–1054 (ACM, New York, 2014).
10. Sauermann, H. & Franzoni, C. *Proc. Natl. Acad. Sci. USA* **112**, 679–684 (2015).
11. Hitlin, P. *Research in the Crowdsourcing Age, A Case Study* (Pew Research Center, Washington, DC, 2016).
12. Bruggemann, J., Lander, G. C. & Su, A. I. *bioRxiv* Preprint at https://www.biorxiv.org/content/early/2017/11/15/220145 (2017).
13. Galton, F. *Nature* **75**, 450–451 (1907).
14. Ipeirotis, P. G., Provost, F. & Wang, J. in *Proc. ACM SIGKDD Workshop on Human Computation* 64–67 (ACM, New York, 2010).
15. Zhai, S., Kong, J. & Ren, X. *Int. J. Hum. Comput. Stud.* **61**, 823–856 (2004).
16. Ipeirotis, P. G. *XRDS* **17**, 16–21 (2010).
17. Scharrel, L., Ma, R., Schneider, R., Jülicher, F. & Diez, S. *Biophys. J.* **107**, 365–372 (2014).
18. Sadanandan, S. K., Ranefall, P., Le Guyader, S. & Wählby, C. *Sci. Rep.* **7**, 7860 (2017).
19. Xie, W., Noble, J. A. & Zisserman, A. *Comput. Methods Biomech. Biomed. Eng. Imaging Vis.* **6**, 283–292 (2016).
20. Vedaldi, A. & Lenc, K. in *Proc. 23rd ACM International Conference on Multimedia* 689–692 (ACM, New York, 2015).
21. Talpalar, A. E. et al. *Nature* **500**, 85–88 (2013).

Acknowledgements

We thank A. Paulson, O. Kiehn, C. Krishnamurthy, C. Nelson, and D. Gordon for contributing microscopy images analyzed in Fig. 1, Fig. 2c, Fig. 2d, Supplementary Fig. 3, and Supplementary Fig. 4, respectively. We thank instructors and students of the 2014 Marine Biology Laboratory Physiology course at Woods Hole for assistance with microtubule gliding assays and image collection (especially R. Fischer). We recognize L. Bugaj, L. Sanders, C. Nilson, and B. Kawas for critical discussion. This work was funded by the Jane Coffin Childs Memorial Fund (postdoctoral fellowship to A.J.H.), the Department of Defense Breast Cancer Research Program (grants W81XWH-10-1-1023 and W81XWH-13-1-0221 to Z.J.G.), the NIH Common Fund (grant DP2 HD080351-01 to Z.J.G.), the NSF (grant MCB-1330864 to Z.J.G.), the UCSF Program in Breakthrough Biomedical Research and the UCSF Center for Cellular Construction (grant DBI-1548297 to S.K.B., S.B., and Z.J.G.), and an NSF Science and Technology Center. Z.J.G. is a Chan Zuckerberg Biohub Investigator.

Author contributions

A.J.H., J.D.M., L.E.B., A.R., and Z.J.G. designed Quanti.us concepts and features. J.D.M. developed and implemented the website and platform. A.J.H., L.E.B., and D.P.B. developed pre- and post-processing code and analyzed raw Quanti.us data. S.K.B. and S.B. developed and implemented the machine learning analysis. All authors wrote and edited the manuscript.

Competing interests

J.D.M. holds an equity interest in Quanti.us LLC. Quanti.us passes payments from users to Amazon Mechanical Turk, which then distributes these payments to workers.

Additional information

Supplementary information is available for this paper at https://doi.org/10.1038/s41592-018-0069-0.

Reprints and permissions information is available at www.nature.com/reprints.

Correspondence and requests for materials should be addressed to Z.J.G.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Methods

Annotation collection using Quanti.us. Image sets were uploaded to quanti.us, a publicly available website developed for this work. The website enables a researcher to upload an image set (each image is considered a task within the larger job), select an annotation tool, provide instructions to Turkers (Supplementary Fig. 6), and specify the desired number of 'replicates' (number of independent Turkers making annotations on each image task). A cost calculator gives the researcher a transparent estimate of the cost of the job before it is submitted. Each image task in the job is created by Quanti.us as a 'human intelligence task' on Amazon Mechanical Turk[16]. When all tasks are complete, Quanti.us returns annotations as Cartesian coordinates, along with individual anonymized Turker identification numbers. Time stamps are also returned for each submitted image, and for each annotation relative to the first annotation made by a given Turker on that image.

Quanti.us can also be used in a free test mode that has two key uses. First, it enables users to test-drive their own job as if they were Turkers, which allows them to check the rendering of their images, how their annotation tool operates, how their instructions appear to Turkers, the format of the raw annotation data they can expect to be returned to them, etc. Second, it can provide a public link to the job that the user can disseminate to any other annotators from other communities such as classrooms, the general public, or their own research group.

We encourage users to experiment with small trial batches of five to ten images before submitting larger batches intended for final analysis in order to determine a set of instructions that best informs Turkers of the intended annotation outcome. Users can also 'stress test' their annotation experiment by sending a link to the Quanti.us test mode version of the job to non-expert peers. These peers can then provide feedback on image quality, instruction clarity, and overall difficulty of the task, for example.

A detailed Matlab pipeline with instructions and example images that covers automated image pre-processing for upload to Quanti.us and many annotation post-processing and overlay options is publicly available (see "Code availability")

Annotation post-processing. Annotations were overlaid onto images and post-processed with custom scripts in Matlab R2015b (Mathworks, Natick, MA). Descriptions of annotation methods by figure are presented in Supplementary Table 1. Spatial subtractive clustering was performed on individual annotations made with the crosshair tool, or on centroids of sets of annotations made with the polyline tool (subclust.m). In analyses of crosshair annotations made by individual Turkers, false positive annotations (fp) were taken as those more than x pixels from the nearest annotation in the corresponding expert ground truth dataset (true positives (tp)), and false negative annotations (fn) were taken as ground truth annotations more than x pixels from the nearest annotation made by the Turker. The value of x was set to twice the average full width at half-maximum of the objects being annotated. The collective performance of Turkers was evaluated via similar computations for clusters rather than for individual annotations. A simple Turker score was defined as $1 - ((\mathrm{fp}' + k \times \mathrm{fn}')/(2 \times \mathrm{tp}'))$, where fp' and fn' are the numbers of false positives and false negatives determined for the Turker under consideration relative to Turker annotation clusters (tp') rather than the expert ground truth annotations. The score can be tailored to specific job types through the arbitrary parameter k (we set $k = 0.2$ for Fig. 1). More complex inherent worker quality scores have also been defined[14].

Spatial r.m.s. errors for each Turker were computed from the minimal distances between their true positive annotations and corresponding ground truth annotations. Performance metrics were precision ($\mathrm{tp}/(\mathrm{tp} + \mathrm{fp})$) and recall ($\mathrm{tp}/(\mathrm{tp} + \mathrm{fn})$). Fitts's law was fit using the Levenberg–Marquardt algorithm[22] (fit.m) as $t = a + b\log_2((c + \sigma)/\sigma)$, where a, b, and c are fitting parameters, and σ is the r.m.s. error.

The centroids of polyline objects associated with each image were clustered and an arbitrary distance threshold was used to determine the membership of a given polyline in a consensus group describing a putative image structure. Polylines outside this threshold were discarded. For microtubule polylines, centroids were computed from the set of annotations in each group and passed as input to FIJI's TrackMate[23] (National Institutes of Health, Bethesda, MD, USA) to determine velocity distributions. The annotations in each group could also be fit by Deming regression[24] to extract consensus polylines and their length distributions (deming.m). For polygon and freehand annotations, we used a threshold on the spatial overlap of outlines to assign them to local consensus groups in each image. We converted outlines in each group to a consensus outline by summing their associated masks and performing thresholding, erosion, and dilation. Overlapping consensus outlines within an image were discarded via a similar thresholding step. For 3D image stacks, a threshold on the overlap between consensus outlines in successive z frames was used to 'connect' annotations to form 3D segmentations.

The MINS analysis[3] in Fig. 2b was conducted for 18 parameter sets comprising all combinations of expected nucleus diameter (20, 30, 40 µm), noise level (2, 3), and kernel smoothing (1.0, 1.5, 2.0).

Users should undertake their own data quality assessment for each job type to ensure interpretability and accuracy of the raw Quanti.us output. This typically involves, first, overlaying raw annotations onto the input images as a visual check for a rough correspondence between image features and annotations. Second, spatial clustering of Turker annotations should be performed, and the results of the Turker collective should be compared with corresponding expert annotations for a small, representative subset of each batch of images submitted. Qualitative or quantitative evaluation of precision and recall metrics is suited to this. The user can generate these expert annotations with the image analysis software of their choice—for example, FIJI—or through the Quanti.us test mode that provides users with a link to a test area where they can annotate their own image sets (see "Annotation collection using Quanti.us").

Machine learning. We trained a machine learning system to predict the center locations of cell nuclei in 500×500 pixel images. Conventional approaches such as the extraction of regional features[25] to develop a region-level detector did not offer viable options owing to the small size of the cellular objects. In contrast, a convolutional neural network (CNN) allowed an end-to-end system design without extraction of separate features to be fed into the learning system. The CNN transforms the image channels by applying a set of 'learnable' filters, successively, directly into an output matrix (as big as the input image) containing high scores in the locations of nuclei centers and very small to almost zero scores elsewhere. The objective of a fully convolutional regression network[19] is to regress this Gaussian weighted output matrix from the input image channels. Here, the term "fully convolutional" refers to the fact that the target variable is a matrix of full image size instead of a vector quantity. The output matrix yields the center locations of the nuclei upon local thresholding. The hierarchical (layered) filter structure constitutes a 'deep' neural net. The filters encapsulate both linear (convolution) and nonlinear (rectified linear unit (ReLU)[26]) operations. We adapted an elegant regression framework for cellular object detection[19]. Further, we implemented a simple CNN architecture that minimizes an L2 loss with an exponentially decreasing learning rate[20,27].

For effective stabilization of the network weights (to avoid overfitting), the training process proceeded in two stages[19]: a pre-training stage (with cropped 100×100 pixel images and augmented by geometric transformations such as flipping and rotation) followed by final training with full images. The test set had five images annotated by six experts. The images were further augmented (by rotation) to make a final test set of 20 images in total.

The deep CNN had layers comprising convolutional kernels and ReLUs. The CNN layers, along with all parameters, were specified in the following order from the input channels to the output[20,27]: convolution ($3 \times 3 \times 2 \times 32$ kernels), ReLU, convolution ($3 \times 3 \times 32 \times 32$ kernels), ReLU, convolution ($3 \times 3 \times 32 \times 1$ kernel). The convolution kernels were initialized with Xavier weights[28]. The peaks of the Gaussian weights in the target matrix were set at 7, in accordance with prior convention[19]. The learning rate started at 10^{-3} for pre-training and at 10^{-2} for final training, and decayed exponentially. The weight decay was set at 10^{-3} for both cases. The scores were thresholded at 1.0, with a window for non-maximum suppression as large as 25×25 pixels.

The center locations of image objects from the thresholded output matrix were scored for false positives and negatives against the expert ground truth as described in the section "Annotation post-processing."

Reporting Summary. Further information on experimental design is available in the Nature Research Reporting Summary linked to this article.

Code availability. Source code is available at https://github.com/quantius-science/. This code is published under the open source MIT license. Researchers are free to use it without restriction. This repository includes code for the Quanti.us pre- and post-processing pipelines and machine learning pipeline.

Data availability. Raw data are available on request from the corresponding author.

References

22. Marquardt, D. W. *J. Soc. Ind. Appl. Math.* **11**, 431–441 (1963).
23. Tinevez, J. Y. et al. *Methods* **115**, 80–90 (2017).
24. Adcock, R. J. *Anal. (Lond.)* **5**, 53 (1878).
25. Arteta, C., Lempitsky, V., Noble, J. A. & Zisserman, A. *Med. Image Anal.* **27**, 3–16 (2016).
26. LeCun, Y., Bengio, Y. & Hinton, G. *Nature* **521**, 436–444 (2015).
27. Vedaldi, A. & Fulkerson, B. in *Proc. 18th ACM International Conference on Multimedia* 1469–1472 (ACM, New York, 2010).
28. Glorot, X. & Bengio, Y. in *Proc. 13th International Conference on Artificial Intelligence and Statistics* (eds. Teh, Y. W. & Titterington, M.) 249–256 (MLR Press/Microtome Publishing, Brookline, MA, 2010).

nature research

Corresponding author(s): Zev J. Gartner

☐ Initial submission ☒ Revised version ☐ Final submission

Life Sciences Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form is intended for publication with all accepted life science papers and provides structure for consistency and transparency in reporting. Every life science submission will use this form; some list items might not apply to an individual manuscript, but all fields must be completed for clarity.

For further information on the points included in this form, see Reporting Life Sciences Research. For further information on Nature Research policies, including our data availability policy, see Authors & Referees and the Editorial Policy Checklist.

▸ Experimental design

1. Sample size

Describe how sample size was determined.

No statistical methods were used to pre-determine sample size. Each annotation case study was completed by groups of turkers of at least 50, with each image for which results are reported receiving annotations by at least 10 turkers (i.e. 10 replicates). The number of turkers contributing to a job is a dependent variable in our jobs, being affected by factors including the number of tasks in a job, the number of replicates, and the number of tasks a worker contributes before dropping out of a job. The number of replicates was arbitrarily set at 10 or 20 to report upper bounds of turker collective accuracy, except for Figure 1d where it was the independent variable. The total number of annotations analyzed in each experiment was typically in the hundreds to hundreds of thousands.

2. Data exclusions

Describe any data exclusions.

No data were excluded for the case studies we present, except during turker annotation filtering using clustering and intrinsic worker quality metrics. We explicitly state where worker filtering was employed in the manuscript, in all cases where we did it.

3. Replication

Describe whether the experimental findings were reliably reproduced.

All attempts at replication were successful for the case studies we present. We validated turker performance across different tasks with specific quantitative metrics, which were broadly consistent between experiments run on different days (see e.g. SI Figure 2 and 3, where aspects of turker performance metrics were found to be consistent across up to 12 independent jobs).

4. Randomization

Describe how samples/organisms/participants were allocated into experimental groups.

Data was collected in an intrinsically random way. We were blind to the dynamics of turkers entering our image annotation jobs, and did no a priori screening or adjustment of turker allocation between tasks in the manuscript

5. Blinding

Describe whether the investigators were blinded to group allocation during data collection and/or analysis.

Investigators were not blinded during analysis of turker annotation data, but raw results were assembled with consistent procedures for case studies in the manuscript. Expert annotations were completed for each experiment before turker data had been collected (i.e. blind to the annotation patterns returned by turkers), and were not modified or adapted thereafter.

Note: all studies involving animals and/or human research participants must disclose whether blinding and randomization were used.

6. Statistical parameters

For all figures and tables that use statistical methods, confirm that the following items are present in relevant figure legends (or in the Methods section if additional space is needed).

n/a	Confirmed	
☐	☒	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement (animals, litters, cultures, etc.)
☐	☒	A description of how samples were collected, noting whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☐	☒	A statement indicating how many times each experiment was replicated
☒	☐	The statistical test(s) used and whether they are one- or two-sided (note: only common tests should be described solely by name; more complex techniques should be described in the Methods section)
☒	☐	A description of any assumptions or corrections, such as an adjustment for multiple comparisons
☐	☒	The test results (e.g. *P* values) given as exact values whenever possible and with confidence intervals noted
☐	☒	A clear description of statistics including central tendency (e.g. median, mean) and variation (e.g. standard deviation, interquartile range)
☐	☒	Clearly defined error bars

See the web collection on statistics for biologists for further resources and guidance.

Software

Policy information about availability of computer code

7. Software

Describe the software used to analyze the data in this study.

Custom code was written in Matlab R2015b, described in detail in the methods, and made available in a repository (see code availability statement). Other software used: FIJI version 2.00-rc-43/1.50g, TrackMate v2.8.1, FIESTA v1.05.0004. Quanti.us was built with Python 2.7 and Django 1.11.

For manuscripts utilizing custom algorithms or software that are central to the paper but not yet described in the published literature, software must be made available to editors and reviewers upon request. We strongly encourage code deposition in a community repository (e.g. GitHub). *Nature Methods* guidance for providing algorithms and software for publication provides further information on this topic.

Materials and reagents

Policy information about availability of materials

8. Materials availability

Indicate whether there are restrictions on availability of unique materials or if these materials are only available for distribution by a for-profit company.

No unique materials were used.

9. Antibodies

Describe the antibodies used and how they were validated for use in the system under study (i.e. assay and species).

No antibodies were used in the study.

10. Eukaryotic cell lines

a. State the source of each eukaryotic cell line used.

Biological experiments were used only to generate images for case studies characteristic of a given task; the annotation results are not used to make specific biological interpretations in the manuscript.

b. Describe the method of cell line authentication used.

No eukaryotic cell lines were used.

c. Report whether the cell lines were tested for mycoplasma contamination.

No eukaryotic cell lines were used.

d. If any of the cell lines used are listed in the database of commonly misidentified cell lines maintained by ICLAC, provide a scientific rationale for their use.

No eukaryotic cell lines were used.

Animals and human research participants

Policy information about studies involving animals; when reporting animal research, follow the ARRIVE guidelines

11. Description of research animals

Provide details on animals and/or animal-derived materials used in the study.

No animals were used in the study.

Policy information about studies involving human research participants

12. Description of human research participants

Describe the covariate-relevant population characteristics of the human research participants.

Our study does not involve human research subjects.